This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (Currently amended) A method for communicating between graphics memory storage elements for performing at least one single instruction, multiple data (SIMD) instruction, comprising the steps of:

(a) reading a set of data into a set of graphics memory storage elements wherein one or more of a plurality of channels have been dedicated for data storage;

(b) identifying, for each graphics memory storage element in said set of graphics memory storage elements, an address wherein said identified address can comprise one or more dimensions;

(c) determining a set of specific data needed for said at least one SIMD instruction;

(d) recalling each said identified address for each graphics memory storage element where said specific data is stored; and

~~(d)~~ (e) using said recalled addresses to retrieve said specific data into another set of graphics memory storage elements wherein one or more of a plurality of channels have been dedicated for data storage.

2. (Original) A method for executing single instruction, multiple data (SIMD) instructions using graphics technology, comprising the steps of:

(a) reading a set of data into a set of texels wherein one or more of a plurality of channels have been dedicated for data storage;

(b) identifying, for each said texel in said set of texels, an address wherein said address may comprise one or more dimensions;

(c) receiving a set of SIMD instructions;

(d) translating said set of SIMD instructions into a set of graphics API commands;

(e) distinguishing a set of specific data, from said set of data, needed for said set of SIMD instructions;

(f) recalling each said identified address for each said texel wherein said specific data is stored;

(g) selecting frame buffer pixels to be used to support said set of SIMD instructions;

(h) using said recalled addresses to retrieve said specific data into said selected frame buffer pixels wherein one or more of a plurality of channels have been dedicated for data storage; and

(i) performing said set of SIMD instructions on said retrieved set of specific data in said selected frame buffer pixels.

3. (Original) The method according to claim 2, further comprising, in place of step (a), the step of:

(a) copying a set of data, stored in a set of frame buffer pixels, into a set of texels wherein one or more of a plurality of channels have been dedicated for data storage.

4. (Original) The method according to claim 3, further comprising, prior to step (a), the step of reading a set of data into a set of frame buffer pixels wherein one or more of a plurality of channels have been dedicated for data storage.

5. (Original) The method according to claim 2, further comprising the step of storing said recalled addresses in said selected frame buffer pixels wherein each dimension of said

recalled addresses is stored in a different dedicated channel and wherein each said stored address defines a particular texel from which a particular piece of data, from said set of specific data, will be retrieved.

6. (Original) The method according to claim 2, further comprising the step of storing said recalled addresses in selected texels in a second texture memory wherein each dimension of said recalled addresses is stored in a different dedicated channel and wherein each said stored address defines a particular texel from which a particular piece of data, from said set of specific data, will be retrieved.

7. (Original) The method according to claim 6, further comprising, after the step of storing, the step of retrieving said stored addresses from said selected texels in said second texture memory to said selected frame buffer pixels.

8. (Original) The method according to claim 2, further comprising the step of storing results of said performed set of SIMD instructions in same said selected frame buffer pixels.

9. (Original) The method according to claim 2, wherein software used to support the method is a graphics application programming interface.

10. (Original) The method according to claim 9, wherein said graphics application programming interface is OpenGL with a pixel texture extension.

11. (Currently Amended) A system for communicating between graphics memory storage elements for performing at least one single instruction, multiple data (SIMD) instruction, comprising:

(a) means to receive a set of data into a set of graphics memory storage elements wherein one or more of a plurality of channels have been dedicated for data storage;

(b) means to identify, for each graphics memory storage element in said set of graphics memory storage elements, an address wherein said identified address can comprise one or more dimensions;

(c) means to determine a set of specific data needed for said at least one SIMD instruction;

(d) means to recall each said identified address for each graphics memory storage element where said specific data is stored; and

~~(d)~~ (e) means to use said recalled addresses to retrieve said specific data into another set of graphics memory storage elements wherein one or more of a plurality of channels have been dedicated for data storage.

12. (Original) A system for executing single instruction, multiple data (SIMD) instructions using graphics technology, comprising:

(a) means to receive a set of data into a set of texels wherein one or more of a plurality of channels have been dedicated for data storage;

(b) means to identify, for each said texel in said set of texels, an address wherein said address may comprise one or more dimensions;

(c) means to receive a set of SIMD instructions;

(d) means to translate said set of SIMD instructions into a set of graphics API commands;

(e) means to distinguish a set of specific data, from said set of data, needed for said set of SIMD instructions;

(f) means to recall each said identified address for each said texel wherein said specific data is stored;

(g) means to select frame buffer pixels to be used to support said set of SIMD instructions;

(h) means to use said recalled addresses to retrieve said specific data into said selected frame buffer pixels wherein one or more of a plurality of channels have been dedicated for data storage; and

(i) means to perform said set of SIMD instructions on said retrieved set of specific data in said selected frame buffer pixels.

13. (Original) The system according to claim 12, further comprising means to copy a set of data, stored in a set of frame buffer pixels, into a set of texels wherein one or more of a plurality of channels have been dedicated for data storage.

14. (Original) The system according to claim 13, further comprising means to read a set of data into a set of frame buffer pixels wherein one or more of a plurality of channels have been dedicated for data storage.

15. (Original) The system according to claim 12, further comprising means to store said recalled addresses in said selected frame buffer pixels wherein each dimension of said

recalled addresses is stored in a different dedicated channel and wherein each said stored address defines a particular texel from which a particular piece of data, from said set of specific data, will be retrieved.

16. (Original) The system according to claim 12, further comprising means to store said recalled addresses in selected texels in a second texture memory wherein each dimension of said recalled addresses is stored in a different dedicated channel and wherein each said stored address defines a particular texel from which a particular piece of data, from said set of specific data, will be retrieved.

17. (Original) The system according to claim 16, further comprising means to retrieve said stored addresses from said selected texels in said second texture memory to said selected frame buffer pixels.

18. (Original) The system according to claim 12, further comprising thereafter means to store results of said performed set of SIMD instructions in same said selected frame buffer pixels.

19. (Original) The system according to claim 1, wherein software used to support the system is a graphics application programming interface.

20. (Original) The system according to claim 12, wherein said graphics application programming interface is OpenGL with a pixel texture extension.

21. (Original) A system for communicating between graphics memory storage elements, comprising:

(a) a texture memory for receiving a set of data into a set of texels wherein one or more of a plurality of channels have been dedicated for data storage;

(b) an address calculator for identify, for each texel in said set of texels, an address wherein said identified address can comprise one or more dimentions;

(c) a frame buffer for storing each said identified address, for each texel where specific data is stored, into a selected set of pixels wherein one or more of a plurality of channels have been dedicated for address storage; and

(d) a pixel-to-pixel communicator for using said stored addresses to retrieve said specific data into said selected set of pixels wherein one or more of a plurality of channels have been dedicated for data storage.

22. (Original) A system for communicating between graphics memory storage elements, comprising:

(a) a frame buffer for receiving a set of data into a set of pixels wherein one or more of a plurality of channels have been dedicated for data storage;

(b) an address calculator for identifying, for each pixel in said set of pixels, an address wherein said identified address can comprise one or more dimensions;

(c) a texture memory for storing each said identified address, for each pixel where specific data is stored, into a selected set of texels wherein one or more of a plurality of channels have been dedicated for data storage; and

(d) a pixel-to-pixel communicator for using said stored addresses to retrieve said specific data into said selected set of texels wherein one or more of a plurality of channels have been dedicated for address storage;

23. (Original) A system for executing single instruction, multiple data (SIMD) instructions using graphics technology, comprising:

(a) a texture memory for receiving a set of data into a set of texels wherein one or more of a plurality of channels have been dedicated for data storate;

(b) an address calculator for identifying, for each said texel in said set of texels, an address wherein said address may comprise one or more dimensions;

(c) a SIMD graphics API translator for receiving a set of SIMD instructions, for translating said set of SIMD instructions into a set of graphics API commands, for distinguishing a set of specific data, from said set of data, needed for said set of SIMD instructions, and for selecting frame buffer pixels to be used to support said set of SIMD instructions;

(d) a frame buffer for storing each said identified address, for each texel where specific data is stored, into a selected set of pixels wherein one or more of a plurality of channels have been dedicated for address storage;

(e) a pixel-to-pixel communicator for using said stored addresses to retrieve said specific data into said selected set of pixels wherein one or more of a plurality of channels have been dedicated for data storage; and

(f) a graphics accelerator for performing said set of SIMD instructions on said retrieved set of specific data in said selected frame buffer pixels.